

Chin C, Si JT, Clare AS, Ma MD. [Intelligent Image Recognition System for Marine Fouling using Softmax Transfer Learning and Deep Convolutional Neural Networks](#). *Complexity* 2017, 2017, 5730419.

Copyright:

© 2017 C. S. Chin et al. This is an open access article distributed under the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

DOI link to article:

<https://doi.org/10.1155/2017/5730419>

Date deposited:

06/11/2017



This work is licensed under a [Creative Commons Attribution 4.0 International License](#)

Research Article

Intelligent Image Recognition System for Marine Fouling Using Softmax Transfer Learning and Deep Convolutional Neural Networks

C. S. Chin,¹ JianTing Si,^{1,2} A. S. Clare,³ and Maode Ma⁴

¹Faculty of Science, Agriculture and Engineering, Newcastle University Singapore, Newcastle upon Tyne, NE1 7RU, UK

²PACC Offshore Services Holdings (POSH) Ltd., No. 1 Kim Seng Promenade, No. 06-01 Great World City, Singapore 237994

³School of Natural and Environmental Sciences, Newcastle University Singapore, Newcastle upon Tyne NE1 7RU, UK

⁴School of Electrical & Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798

Correspondence should be addressed to C. S. Chin; cheng.chin@ncl.ac.uk

Received 14 July 2017; Accepted 10 September 2017; Published 15 October 2017

Academic Editor: Jing Na

Copyright © 2017 C. S. Chin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The control of biofouling on marine vessels is challenging and costly. Early detection before hull performance is significantly affected is desirable, especially if “grooming” is an option. Here, a system is described to detect marine fouling at an early stage of development. In this study, an image of fouling can be transferred wirelessly via a mobile network for analysis. The proposed system utilizes transfer learning and deep convolutional neural network (CNN) to perform image recognition on the fouling image by classifying the detected fouling species and the density of fouling on the surface. Transfer learning using Google’s Inception V3 model with Softmax at last layer was carried out on a fouling database of 10 categories and 1825 images. Experimental results gave acceptable accuracies for fouling detection and recognition.

1. Introduction

Marine biofouling is the unwanted colonisation and growth of marine organisms on immersed artificial structures [1]. Following rapid conditioning of the surface, subsequent fouling is a dynamic process, which depends on availability, the relation of colonisers to incumbents, and the speed at which organisms can attach [2], though it is often described as successional [3]. With regard to shipping, the main issue for hull performance is an increase in frictional resistance, which requires increased power to maintain speed and hence increased fuel consumption and gaseous emissions [4, 5]. Hull fouling is also the main vector for translocation of nonindigenous species [6]. The early detection and recognition of marine growth are therefore paramount to avoid these problems. This paper studies marine growth detection and recognition using the transfer learning and deep convolutional neural network (CNN) approach.

Artificial Intelligence (AI) uses machine learning where the test data are extracted from a similar n -dimensional

space. In most practical applications, an immense amount of computational resource and capital is needed to obtain the required dataset to reconstruct the model [7]. As a result, transfer learning is used. It can reduce the need for gathering resources and capital and at the same time use the pretrained model from a different application to work on the current application. Transfer learning uses learning from a preceding training model. Features learned by a CNN model originally trained on a huge dataset can be used to perform recognition tasks in a particular domain of the dataset [8–10]. Several methodologies for transfer learning have produced good results and there is a large literature on transfer learning and CNN. For example, Devikar [11] adopted transfer learning on 11 types of dog breeds, with 25 images for each class. He fine-tuned after retraining Google’s Inception V3 model from the dog dataset and achieved 96% accurate identification of the dog breed. The utility of the approach has also been demonstrated for plant images [12, 13], medical datasets [14–16], and a State Farm dataset [17] with an accuracy of up to 98% in the case of the Tapas [13]

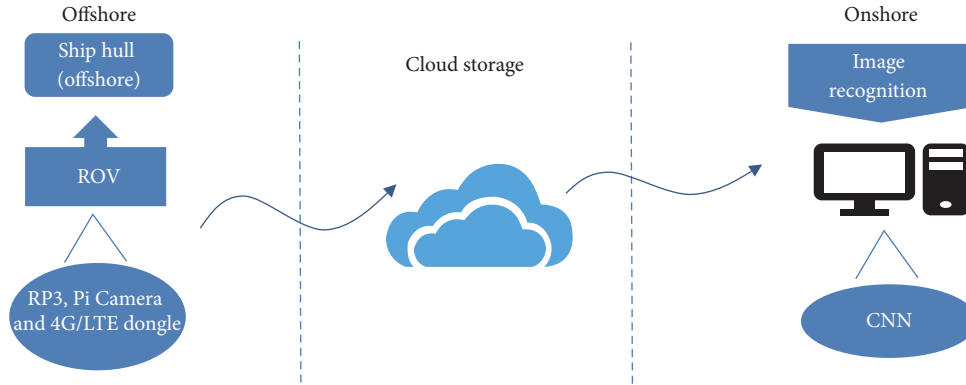


FIGURE 1: Proposed fouling recognition system design in this study.

study. In all these cases, a large database of good quality images with strong and distinctive features was required to produce good classification results. The transfer learning removes the need of a Graphics Processing Unit (GPU) for training despite the advantage of shortening the training time. The transfer learning surpasses full training of a CNN model on classification accuracy. It also reduces the need to label data by using features already learned from the previous model.

Although this approach has also been applied to water filtration and thermal investigations of fouling, for example, to predict fouling in the filtration of drinking water [18], heat exchangers [19], and heaters [20], transfer learning and deep CNN have not been applied to biofouling of marine structures such as ships' hulls. Notably, the above-mentioned classification results yielded more than 70% accuracy after using the transfer learning approach on CNN. This value is used as a benchmark for the present study, which is focussed on deploying transfer learning using Google's Inception V3 model [21, 22] with Softmax on a fouling database with ten categories and 1825 images. Retraining of the fouling database was performed using TensorFlow [23, 24] Docker image deep learning library.

In summary, the application of the Softmax transfer learning and deep convolutional neural networks on marine fouling recognition has not been reported in the literature. The early detection of marine growth through image recognition provides a means for the vessel's owner to schedule maintenance, for example, by hull grooming [25, 26], before the marine growth seriously compromises hull performance. Hence, the contributions made by the paper are as follows. The images of fouling taken from the ship can be transferred remotely to onshore for fouling recognition. The transfer learning and deep convolutional neural network can be used to classify the types and density of fouling using the captured image for early detection of fouling.

The remaining sections are organized as follows. Section 2 describes the fouling image recognition system followed by the proposed deep convolutional neural network in Section 3. Lastly, Section 4 concludes the paper.

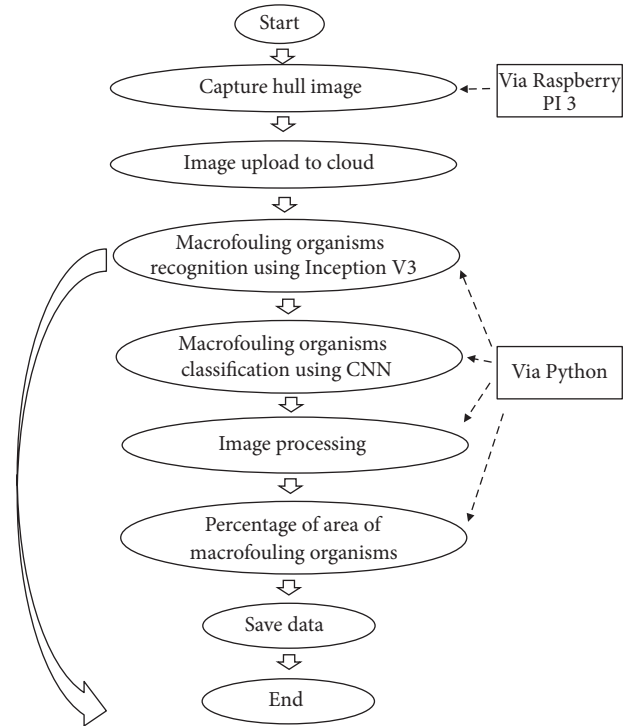


FIGURE 2: Software program flowchart.

2. Fouling Image Recognition System

The project aims to take a picture of a particular area on the surface of a ship hull via the camera module in the microcontroller. Due to field testing constraints, the captured images were obtained from the web. Some of the images do not truly represent the actual fouling on ships' hulls. The captured image will be stored in cloud storage via 4G/Long-Term Evolution (LTE) dongle. It will be automatically uploaded to the Cloud in Figure 1. The image will be used for fouling analysis via Deep Learning and CNN. As shown in Figure 2, the first stage of the program will identify if there are any macrofouling organisms on the



FIGURE 3: Different classes of fouling images used for training.

TABLE 1: Different classes and number of fouling images.

Fouling image class	Image count
Rock oysters	82
Kelp	272
Zebra mussel	220
Hydrozoan	116
Acorn barnacle	83
Algae	195
Finger sponge	119
Gooseneck barnacle	288
Christmas tree worm	222
Tunicates	228

surface of the ship hull using camera in Raspberry Pi. It is performed by carrying out image recognition using the Inception V3 model that was trained for ImageNet Large Scale Visual Recognition Challenge [21]. If the result indicates that there are no macrofouling organisms, the algorithm will stop. The percentage of macrofouling on the surface or fouling density will be determined. Otherwise, it will proceed to the next stage where the macrofouling organisms will be classified according to different classes of marine fouling as seen in Table 1 and Figure 3. There are a total of 1825 images in the database. The classes of marine fouling are

not exhaustive as there are more than 4,000 known fouling species. The fouling in Table 1 was decided by adapting the few common types of fouling on the vessels which allows the model to learn sufficient features for the classifier to perform its task. Due to the inconsistency in image size, quality, and differences in the fouling images, numbers of fouling images for each class are unequal. The images will be refined to be more representative of marine biofouling on ships' hulls and at earlier stages of development. In the last stage, the algorithm will detect the percentage of macrofouling organisms over the total area of the image known as the fouling density. This is achieved by processing the image via Open Source Computer Vision Library (OpenCV) using the color-based segmentation method. The various results obtained throughout the program can be saved. A Graphic User Interface (GUI) was designed to facilitate the process of running the different stages of the algorithms.

The arrangement of the image recognition system for biofouling is shown in Figure 4. It was not possible to arrange the test on board a ship. Instead, a laboratory setup was used. The fouling images were secured to a wall to simulate the presence of the fouling on the ship's hull. The left-hand side shows the microprocessor connected via mobile connection (situated on board the "ship") into cloud storage and to the local machine on the right side. The host device ("onshore") is used for fouling recognition and analysis. The microprocessor (i.e., Raspberry Pi 3 Model B) operates on Raspbian OS (see Figure 5), and the local machine uses

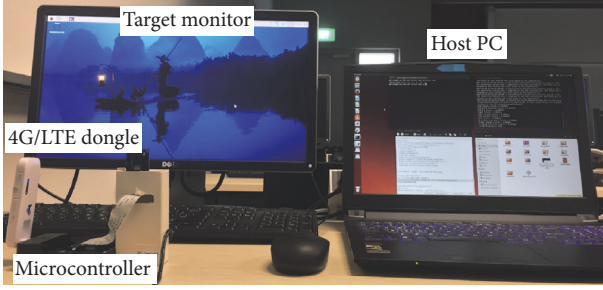


FIGURE 4: Overview of fouling image recognition system setup includes microprocessor connected via mobile connection (left) and host device (right) in laboratory.

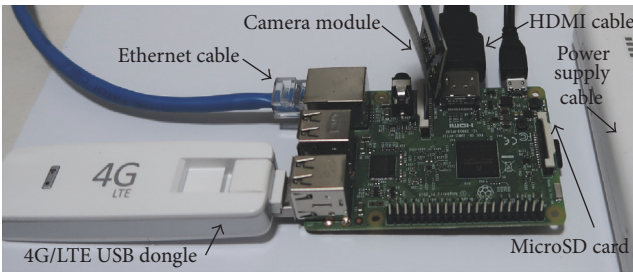


FIGURE 5: Microcontroller connection for fouling image recognition system.

Ubuntu OS. It can function as a computer with a built-in camera for building a smart device like fouling recognition system. For completeness, Tables 2 and 3 show the hardware and software components in the proposed system.

3. Proposed Deep Convolutional Neural Network

Convolutional neural network (CNN) is a type of deep learning neural network (NN) for image recognition and classification. This kind of neural network is made out of layers linked by artificial neurons that form a connection between the layers. The link carries valued weights that are fine-tuned throughout the training process, resulting in a trained network. The layers are constructed such that the first layer identifies a set of simple patterns of the input. A standard CNN contains 5 to 25 such layers and ends with an output layer [27]. Figure 6 shows a typical architecture of CNN used in this paper.

In the first convolution layer, features of inputs are extracted, such as edges, lines, and corners of the image. Taking a 3×3 feature extractor in a $n \times n$ size image, as an example and starting from the top left corner of an image, the feature extractor performs matrix multiplication and addition with the pixel value and then sums them. The summed value will enter the first hidden layer, which is also known as the first feature map. The feature extractor now slides itself by stride equal to 1 to the right and repeats the process. When the feature extractor reaches the far right of

TABLE 2: Hardware components.

Hardware name	Quantity	Remarks
Raspberry Pi 3 Model B	1	Microcontroller
Raspberry Pi camera	1	Image capturing
Raspberry Pi micro USB power supply	1	Power supply
HUAWEI 4G dongle	1	Cellular network dongle
4G LTE SIM card	1	Cellular network provider

TABLE 3: Software components.

Software name	Quantity	Purpose
Raspbian OS	1	OS for RP3
Ubuntu OS 14.04	1	OS for local machine
Python 3	1	Programming tool
TensorFlow Docker image	1	Deep learning library
Spider	1	Code editing tool

the image, it will move vertically down by stride equal to 1, return to the far left of the image, and repeat the process again. This process will continue until the 3×3 feature extractor reaches the bottom right of the image [27].

The pooling layer follows the convolutional layers. Pooling layers build up resistance against noise and distortion in the features. Maximum pooling and average pooling are the two ways to perform this task. Taking the example of a 4×4 feature map input into the pooling layer, for 2×2 pooling, the 4×4 input will be split into four matrices that do not lap over one another. In max pooling, the 2×2 pooling will only consider the largest value within itself to be the output. For average pooling, the 2×2 pooling will consider the average of the four values within itself to be the output. The advantage of pooling in image recognition is that it is invariant to slight shifting and rotating of the image. Therefore, pooling is a process that condenses the convolutional layers [27].

The nonlinear layer is an additional function that is applied after every convolution. The CNN takes advantage of this function by suggesting the prominent classification of likely similar features within the hidden layers. The CNN also uses various nonlinear functions and the most popular function is the rectified linear units (ReLU). The ReLU operates on a function of each pixel and substitutes all negative pixel values in the feature map by 0. The ReLU is needed as convolution is a linear operation and the data introduced into CNN is always nonlinear [27].

The fully connected layer is used as the last layer of CNN. The fully connected layer performs a summation of weights of features from its preceding layers. It implies that high-level features of outputs obtained from previous convolution and pooling are being classified based on the input image and the dataset the CNN is trained on. This classification will be brought over to the output layer where the results will be shown as the probability of result or class probability as seen in Tables 4 and 5.

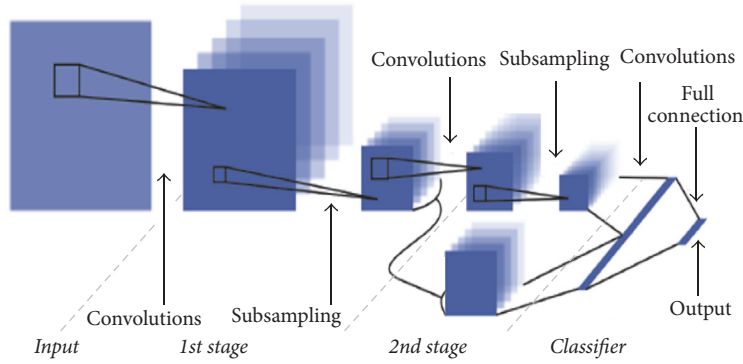


FIGURE 6: Architecture of convolutional neural network [10].

TABLE 4: Classification results of fouling (first and second training).

Classes	First training (%)	Second training (%)	Percent of improvement (%)
Rock oysters	75.945	96.236	20.291
Kelp	83.817	99.490	15.673
Zebra mussel	77.143	98.350	21.207
Hydrozoan	58.841	92.032	33.191
Acorn barnacle	62.001	96.771	34.770
Algae	58.749	89.231	30.482
Finger sponge	39.369	80.767	41.398
Gooseneck barnacle	89.061	99.903	10.842
Christmas tree worm	65.098	95.908	30.810
Tunicates	86.042	96.344	10.302

TABLE 5: Final testing results of fouling classification.

Classes of fouling	Test results (%)
Rock oysters	99.703
Kelp	99.128
Zebra mussel	87.916
Hydrozoan	97.635
Acorn barnacle	77.228
Algae	79.255
Finger sponge	76.617
Gooseneck barnacle	91.430
Christmas tree worm	94.961
Tunicates	99.171

The proposed Inception V3 model comprises three convolutional layers with a pooling layer, the three convolutional layers, ten inception modules, and a fully connected layer. There are a total of 17 layers in this model that contains features learned from the original training with ImageNet. The original fully connected layer will be retrained by a new layer. The activators of the bottleneck layer are used to produce bottleneck values that are placed in the Softmax

classifier. The new Softmax function will map the input image data to obtain the classification results [28].

3.1. Transfer Learning Using Inception V3 Model with Softmax.

The transfer learning that utilizes a pretrained neural network was implemented to identify and recognize the macrofouling organisms during the first stage of the program. This section shows the result of transfer learning using Inception V3 model with Softmax on the fouling image. Instead of training a deep network from scratch, a network trained on a different application is used. In this project, an image recognition model known as Inception V3 [21] was chosen. It consists of two main parts, namely, the feature extraction with a CNN and the classification part with fully connected and Softmax layers. The Inception V3 is capable of achieving good accuracy for image recognition of macrofouling organisms. Before the retraining process, the fouling database is arranged in the labeled directories. The ten classes of the dataset are stored in the directory of "fouling photos." The retraining process in TensorFlow Docker image was executed. The TensorFlow Docker helps to ease the starting and running of the TensorFlow open source software library. The first retraining process runs on 500 training steps with default train batch size of 10 and a learning rate of 0.01. The bottleneck values of each fouling image are stored in the bottleneck

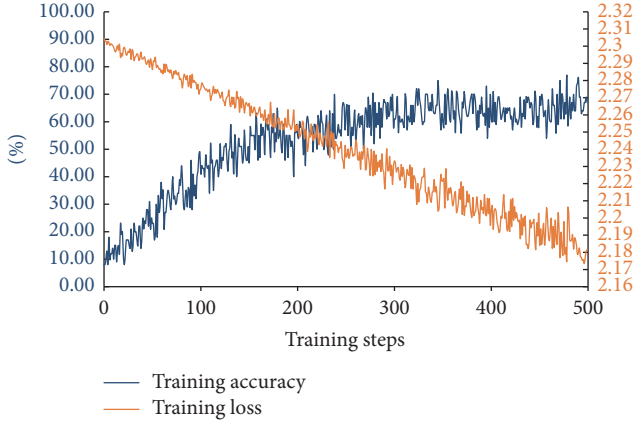


FIGURE 7: Training accuracy and loss for 1st training.

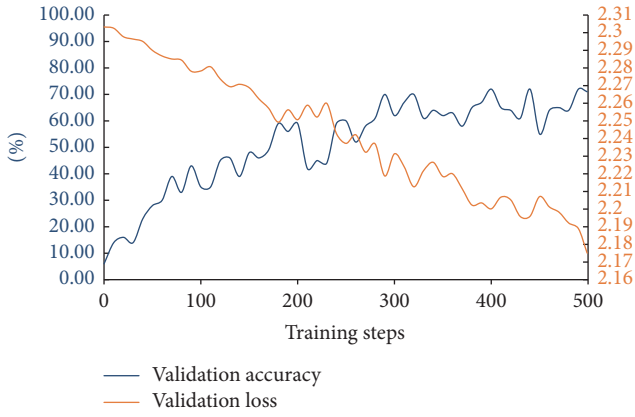


FIGURE 8: Validation accuracy and loss for 1st training.

directory. The predictions are compared with the actual labeled image. The evaluation process will update the final layer's weights through a backpropagation process.

The training accuracy indicates the probability that the fouling images were identified correctly. The validation accuracy equates to the probability that any chosen image was identified correctly. The cross entropy is a function that reflects how far apart the images being classified are from their ground truth labels. A new retrained Inception V3 model graph, retrained labels text file containing ten fouling classes, and retrained log files are produced after the retraining process. The log files map out the training and validation accuracies with their losses. The first retraining process is plotted as shown in Figures 7 and 8. Random fouling photos were chosen from the dataset to test the new Softmax classifier. With the Softmax function, the input image was classified and sorted in order of confidence. As shown in Table 4, some of the classes such as hydrozoan, acorn barnacle, algae, finger sponge, and Christmas tree worm produced a result of less than 70%, which as mentioned above was set as the benchmark for what is acceptable.

Hence, the fouling samples in the fouling database are retrained to improve the validation accuracy. The retraining

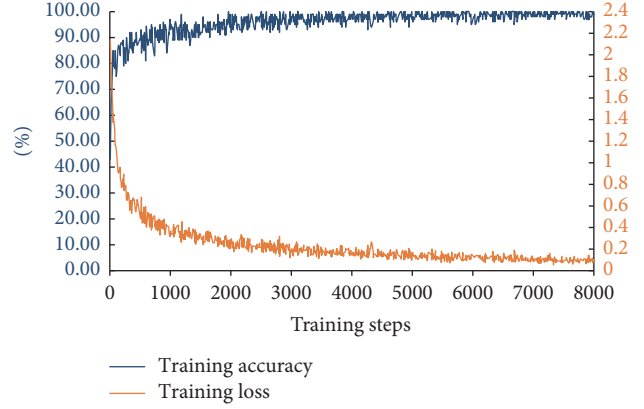


FIGURE 9: Training accuracy and loss for 2nd training.

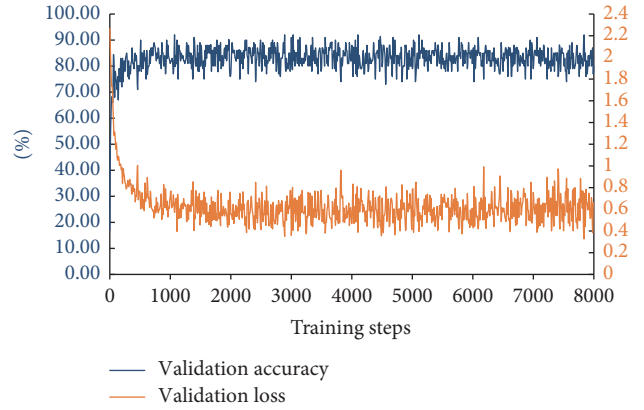


FIGURE 10: Validation accuracy and loss for 2nd training.

process repeats until a higher final test accuracy is obtained. A set of random fouling images were used to test the classifier. The results are tabulated as shown in Table 4. Further retraining iterations were performed with different batch size and learning rate. However, it did not produce a better result. Moreover, the final test accuracy is worse than the first training. It was found that the fouling photos of algae did not possess sharp features. More images of algae were then downloaded for retraining. Some amendments were performed to produce an image of better quality with less noise. The second training was then repeated with the initial batch size and learning rate. Table 4 shows around 10 to 40% improvement in the second training. The final test in Table 5 shows improvement in the classification. For example, Figures 9 and 10 show the results of the second training for algae. The average result improved to 79.255%. Most of the previous methodologies of transfer learning generated classification accuracy of over 70%. In this paper, the classification results produce satisfactory classification results above 70% as shown in Table 5.

3.2. Macrofouling Recognition. The image processing algorithm was then tested on acorn barnacles that are a common type of macrofouling organism. This task of detecting the

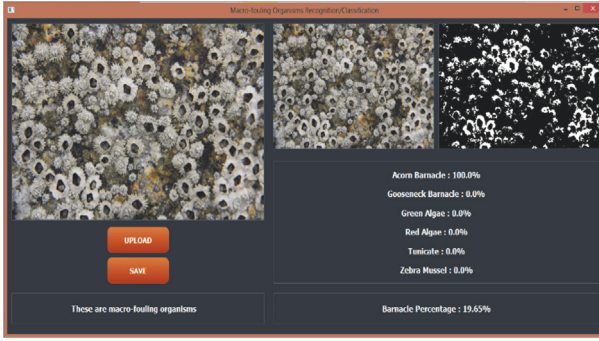


FIGURE 11: Proposed graphical user interface for image fouling recognition.

presence of macrofouling was accomplished via OpenCV. However, the algorithm can be modified to consider another type of fouling. Both the Red; Green; Blue (RGB) and Hue; Saturation; Value (HSV) color spaces have been tested. The RGB color model is more efficient and accurate than HSV. Fouling species of barnacle, especially in the tropics, are commonly red or white with red or purple stripes. But the shells may be colonised or overgrown by other fouling forms, which will complicate recognition/classification. Shapes of the openings of the shells will also vary depending on species besides the color. For a start, the color-based segmentation method was then implemented, and the morphological (shape) operation was carried out to process the image. An RGB lower and upper limit were decided for an off-white color similar to the color of acorn barnacles in the images used. A mask was then used to obtain the desired barnacle color. The image was then converted into greyscale. The morphological closing operation was used to close small holes inside the foreground objects. The number of 1 (i.e., white) that represents the acorn barnacle pixels was determined as shown in Figure 11. The total area of the image was also computed. The number of acorn barnacle pixels was divided over the entire area of the image to obtain the percent of acorn barnacle fouling. The routine used in this project relies on the CNN, and the accuracy depends on the size of the dataset for training.

A GUI is developed to allow users to interact with the functions as seen in Figure 11. The GUI was developed using PyQt5 to facilitate the flow of the algorithm, namely, recognition of macrofouling organisms by OpenCV, classification of macrofouling organisms via CNN model, and lastly the percent of acorn barnacle fouling. The GUI allows the users to upload the image via the “UPLOAD” button. The image and results of the classification and fouling density will be displayed as shown in Figure 12. If the image uploaded is nonmacrofouling organisms, the function buttons will be disabled. The results can be saved via the “SAVE” button to allow users to save the results for further analysis. Around 60 positive images (10 from each class shown in Figure 11) and 60 negative images (any other images) with size 720×480 in dimension were obtained from the Internet to test the accuracy of the recognition. Of the 60 positive images

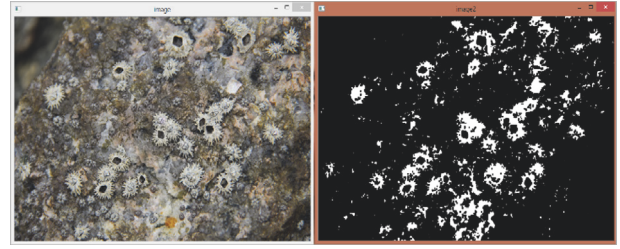


FIGURE 12: Acorn barnacle percent fouling over the total area.

tested, 53 images were correctly identified as macrofouling organisms while seven images were wrongly identified as nonmacrofouling organisms, giving an accuracy of 88%. Of the 60 negative images tested, 51 images were correctly identified as nonmacrofouling organisms while nine images were wrongly identified as macrofouling organisms, giving an accuracy rate of 85%. The mean accuracy for this experiment is, therefore, 86.50%, which is relatively high given that a pretrained model was used. The results show that the image processing algorithm can detect the presence of macrofouling organisms.

3.3. Macrofouling Classification by CNN. Further simulations were carried out on two different datasets to determine whether the accuracy will be affected by the different number of images for each class. In this study, the quality and relevance of the images are acceptable for training. Dataset #1 contained a total of 1423 images, with 997 and 426 containing the training and validating images, respectively. Dataset #2 contained a total of 582 images, with 402 and 180 being training and validating images, respectively. For both datasets, epochs (or a number of steps) were set at 100/75/50/25, respectively. As seen from Table 6, the accuracy increases with the number of epochs and the size of the dataset. A larger number of epochs and datasets should be used to train the CNN model to improve the training accuracy. A larger network with a greater number of layers can be implemented to improve the accuracy. However, the extensive neural network will cause severe overfitting and will be computationally expensive to train.

After training the CNN models, approximately 60 positive images used in previous training were used to test the different trained models to compare their accuracies in classification. As observed in Table 7, the accuracy increases with the number of epochs. The classification of the macrofouling organisms gave a mean accuracy of 74.75%, median of 70.50%, and standard deviation of 7.92%.

3.4. Macrofouling Density. The acorn barnacle images were used to obtain the percent fouling over the total area of the image or the fouling density by OpenCV. Around 40 acorn barnacle images were used for the test. The results obtained gave the lowest fouling density of 8.58%, highest value of 36.79%, and mean of 20.18%. Based on the results obtained, the fouling density will be categorized. For example, a range of 5%–15% will be considered light fouling, 15%–25%

TABLE 6: CNN models training results.

Data	Number of images	Training images	Validating images	Model name	Epochs	Batch size	Loss	Accuracy	Val. loss	Val_acc
Dataset 1	1423	997	426	CNN_Model.h1	100	96	0.0166	0.9354	0.0803	0.6879
				CNN_Model.h2	75		0.0278	0.8864	0.0743	0.6848
				CNN_Model.h3	50		0.0475	0.8062	0.0902	0.6303
				CNN_Model.h4	25		0.0669	0.7206	0.0673	0.7091
Dataset 2	582	402	180	CNN_Model.h5	100	96	0.0191	0.9225	0.1046	0.6250
				CNN_Model.h6	75		0.0242	0.9182	0.1041	0.6250
				CNN_Model.h7	50		0.0302	0.8916	0.0670	0.6771
				CNN_Model.h8	25		0.0617	0.7351	0.0836	0.6146

TABLE 7: CNN models test results.

Data	Model name	Epochs	Batch size	Acorn barnacles	Gooseneck barnacle	Algae	Tunicate	Zebra mussel	Accuracy
Dataset 1	CNN_Model.h1	100	96	6/10	10/10	10/10	9/10	9/10	89%
	CNN_Model.h2	75		8/10	10/10	9/10	9/10	6/10	87%
	CNN_Model.h3	50		10/10	6/10	8/10	7/10	6/10	69%
	CNN_Model.h4	25		8/10	7/10	8/10	9/10	1/10	67%
Dataset 2	CNN_Model.h5	100	96	4/10	8/10	7/10	10/10	4/10	72%
	CNN_Model.h6	75		8/10	6/10	8/10	10/10	3/10	69%
	CNN_Model.h7	50		9/10	6/10	8/10	9/10	6/10	78%
	CNN_Model.h8	25		9/10	2/10	9/10	9/10	3/10	67%

medium fouling, and 25%–40% heavy fouling. The routine is capable of determining the percentage of fouling and the level of fouling as shown in Figure 12. The test for the acorn barnacle can be modified to account for the different classes by defining the RGB color range based on their color feature. However, there are some limitations for the tunicate class as they exist in many different colors and morphologies. In summary, the transfer learning using Google's Inception V3 model with Softmax at last layer was successfully performed on a fouling database of 10 categories and 1825 images. The fouling was classified correctly with over 70% validation accuracy. The image processing approach can also detect the percent of fouling on a given surface area of interest.

4. Conclusion

Marine biofouling has adverse effects on marine vessel operational performance and costs. The rapid and low-cost implementation of image recognition through classification via transfer learning on a pretrained convolutional neural networks (CNN) model (named Inception V3) provides a potential solution for early detection of fouling on ship's hull. The wireless transmission via mobile network enables a fouling image to be uploaded to Google Drive cloud storage easily and subsequently used for image recognition. The graphical user interface was developed with the functions to facilitate the flow of the fouling recognition algorithms from fouling recognition and fouling classification to the density of fouling via CNN and Open Source Computer Vision Library (OpenCV). The acorn barnacle was recognized and

classified with acceptable validation accuracy. The percent of fouling due to acorn barnacle on given surface area was also determined successfully.

This study has the desired benchmark in the final fouling classification accuracy and fouling density. Future work will develop a database of “real-world” images of a greater diversity of fouling organisms at different stages of development and examine the potential of the CNN approach for remote assessment of fouling. Other types of unsupervised machine learning techniques will be used to compare with the proposed CNN approach. The shapes of the openings of the species will be included in macrofouling recognition. The testing of the image recognition system will be conducted on board the ship via a maritime satellite broadband system such as VSAT or SEVSAT.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors would like to express the deepest appreciation to Mr. Goh Jun Yi, Mr. Galvan Goh Jia Xuan, and Mr. Clauson Seah Jie Tai who graduated from Newcastle University for sharing their reports on fouling recognition and classification. The authors would like to acknowledge the Global Excellence Fund Award (ID: ISF#51) from Newcastle University for sponsoring and supporting the project since 2015.

References

- [1] J. A. Lewis, "Marine biofouling and its prevention on underwater surfaces," *Materials Forum*, vol. 22, pp. 41–61, 1998.
- [2] A. S. Clare, D. Rittschof, D. J. Gerhart, and J. S. Maki, "Molecular approaches to nontoxic antifouling," *Invertebrate Reproduction and Development*, vol. 22, no. 1-3, pp. 67–76, 1992.
- [3] M. Wahl, *Marine Epibiosis. 1. Fouling and Antifouling - Some Basic Aspects*, vol. 58, Marine Ecology Progress Series, 1989.
- [4] M. P. Schultz, "Effects of coating roughness and biofouling on ship resistance and powering," *Biofouling*, vol. 23, no. 5, pp. 331–341, 2007.
- [5] M. P. Schultz, J. A. Bendick, E. R. Holm, and W. M. Hertel, "Economic impact of biofouling on a naval surface ship," *Biofouling*, vol. 27, no. 1, pp. 87–98, 2011.
- [6] F. T. Chan, H. J. Macisaac, and S. A. Bailey, "Relative importance of vessel hull fouling and ballast water as transport vectors of nonindigenous species to the canadian arctic," *Canadian Journal of Fisheries and Aquatic Sciences*, vol. 72, no. 8, pp. 1230–1242, 2015.
- [7] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [8] J. Donahue, Y. Jia, O. Vinyals et al., "DeCAF, a deep convolutional activation feature for generic visual recognition," *In Icm1*, vol. 32, pp. 647–655, 2014.
- [9] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2015*, pp. 34–42, usa, June 2015.
- [10] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *Proceedings of the International Joint Conference on Neural Network (IJCNN '11)*, pp. 2809–2813, IEEE, August 2011.
- [11] P. Devikar, "Transfer Learning for Image Classification of various dog breeds," *International Journal of Advanced Research in Computer Engineering and Technology (IJARCET)*, vol. 5, no. 12, pp. 2707–2715.
- [12] A. K. Reyes, J. C. Caicedo, and J. E. Camargo, "Fine-Tuning Deep Convolutional Networks for Plant Recognition," *fra*, September 2015, 1391, 1–9.
- [13] A. Tapas, "Transfer learning for image classification and plant phenotyping," *International Journal of Advanced Research in Computer Engineering and Technology (IJARCET)*, vol. 5, no. 11, pp. 2664–2669.
- [14] H. Shin, H. R. Roth, M. Gao et al., "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.
- [15] N. Bayramoglu and J. Heikkilä, "Transfer Learning for Cell Nuclei Classification in Histopathology Images," *In Computer Vision—ECCV 2016 Workshops*, pp. 532–539, 2016.
- [16] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-scale Image Recognition." arXiv preprint arXiv:1409.1556, 2014.
- [17] D. Singh, "Using Convolutional Neural Networks to Perform Classification on State Farm Insurance Driver Images," 4321–4325.
- [18] N. V. Delgrange, C. Cabassud, M. Cabassud, D. L. Bourlier, and J. M. Laine, "Neural networks for long term prediction of fouling and backwash efficiency in ultrafiltration for drinking water production," *Desalination*, vol. 131, no. 1-3, pp. 353–362, 2000.
- [19] S. Lalot and H. Pálsson, "Detection of fouling in a cross-flow heat exchanger using a neural network based technique," *International Journal of Thermal Sciences*, vol. 49, no. 4, pp. 675–679, 2010.
- [20] S. Lalot and S. Lecoeuche, "Online fouling detection in electrical circulation heaters using neural networks," *International Journal of Heat and Mass Transfer*, vol. 46, no. 13, pp. 2445–2457, 2003.
- [21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, Lake Tahoe, Nev, USA, December 2012.
- [22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 2818–2826, July 2016.
- [23] M. Abadi, P. Barham, J. Chen et al., "TensorFlow, A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI, Savannah, Ga, USA, 2016)*.
- [24] B. J. Erickson, P. Korfiatis, Z. Akkus, T. Kline, and K. Philbrick, "Toolkits and Libraries for Deep Learning," *Journal of Digital Imaging*, pp. 1–6, 2017.
- [25] M. Tribou and G. Swain, "The use of proactive in-water grooming to improve the performance of ship hull antifouling coatings," *Biofouling*, vol. 26, no. 1, pp. 47–56, 2010.
- [26] M. Tribou and G. Swain, "Grooming using rotating brushes as a proactive method to control ship hull fouling," *Biofouling*, vol. 31, no. 4, pp. 309–319, 2015.
- [27] S. Hijazi, R. Kumar, C. Rowen, Group IP, and Cadence, "Using Convolutional Neural Networks for Image Recognition," http://owl.english.purdue.edu/owl/resource/560/01/https://ip.cadence.com/uploads/901/cnn_wp-pdf.
- [28] M. S. Singh, V. Pondenkandath, B. Zhou, P. Lukowicz, and M. Liwicki, "Transforming sensor data to the image domain for deep learning — An application to footprint detection," in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2665–2672, Anchorage, AK, USA, May 2017.

